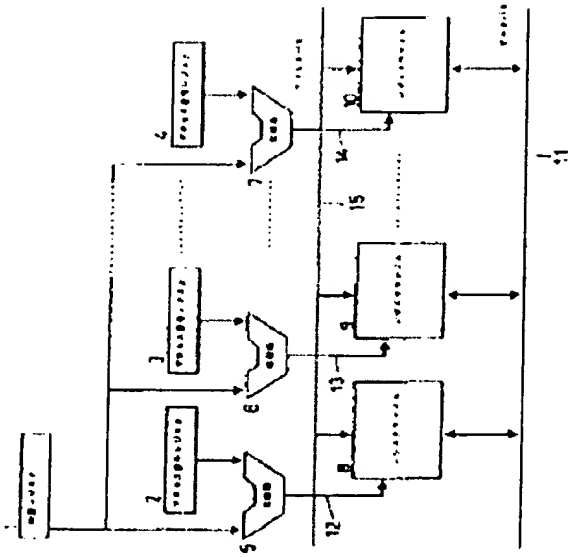
REGISTER CIRCUIT

Patent number: JP3057025
Publication date: 1991-03-12
Inventor: YAMAMOTO TAKAO
Applicant: MATSUSHITA ELECTRIC IND CO LTD
Classification:
- international: G06F9/46
- european:
Application number: JP19890192900 19890725
Priority number(s):

Abstract of **JP3057025**

PURPOSE:To switch a process at a high speed by comaring respective contents of N process number registers with contents of a status register and selecting a register file corresponding to a coincidence detection signal.

CONSTITUTION:N (N>=2) register files 8 to 10 are provided, and N process number register 2 to 4 and N comparators 5 to 7 are provided for these register files 8 to 10 in 1:1. One of register files 8 to 10 corresponding to the process number register 2 to 4 having the same contents as a status register 1 out of process number registers 2 to 4 is accessed. Consequently, contents of the status register 1 are only rewritten with the process number of another process at the time of switching a process to switch the register file 8 to be used. Thus, the process is switched without executing the unnecessary external access.



Data supplied from the ***esp@cenet*** database - Worldwide

BEST AVAILABLE COPY

instruction style ≒ instruction flow

* NOTICES *
Japan Patent Office is not responsible for any
damages caused by the use of this translation.
1.This document has been translated by computer. So the translation may not reflect the original precisely.
2.**** shows the word which can not be translated.
3.In the drawings, any words are not translated.

DETAILED DESCRIPTION

[Detailed Description of the Invention]

[0001]

[Field of the Invention] By publishing the instruction of two or more instruction styles to juxtaposition, this invention reduces the pipeline interlocks by the dependency during an instruction, and relates to the information processor which uses two or more instruction execution means efficiently.

[0002]

[Description of the Prior Art] By executing the instruction of two or more instruction styles to juxtaposition, the use effectiveness of an instruction execution means is gathered and the method of aiming at improvement in processing speed is indicated in JP,6-44089,A and JP,7-182168,A. The outline block diagram for the principal part of the conventional information processor is shown in drawing 3 . For 101,102, as for an instruction decoder and 330, in drawing 3 , a resource allocation means, and 410,420 and 430 are [an instruction fetch unit and 210,220] instruction execution means, respectively.

[0003] In order to publish the instruction of two instruction styles to juxtaposition, this conventional information processor The instruction of two instruction styles is fetched from memory (not shown) using two instruction fetch units 101,102. The instructions 30 and 40

fetched from each instruction style are decoded and analyzed by the instruction decoder 210,220, respectively. The resource allocation means 330 The condition of the information decoded and analyzed by the instruction decoder 210,220 and the instruction execution means 410,420,430 etc. is used. Publishing the instructions 30 and 40 decoded by the instruction decoder 210,220 for the instruction execution means 410,420 or 430, the instruction execution means 410,420 or 430 perform an operation according to the instructions 30 and 40 assigned to each.

[0004] Thereby, the instruction of two or more instruction styles was published for the instruction execution means 410,420,430 at juxtaposition, and improvement in the engine performance is realized.

[0005]

[Problem(s) to be Solved by the Invention] By the advance of semi-conductor accumulation equipment in recent years, the improvement in the internal frequency of a processor is remarkable, and access of external memory requires very many numbers of processor cycles increasingly. Then, what is necessary is to process many instruction styles to coincidence and just to conceal external memory access latency in the above information processors, in order to publish the instruction of two or more instruction styles to coincidence and to reduce the dependencies during an instruction more.

[0006] however, with the above-mentioned configuration, in order to supply the instruction of two or more instruction styles to each functional units (an adder subtracter, branching processing section, load store section, etc.) efficiently [whether the configuration two or more instruction fetch units of whose can perform an instruction fetch to coincidence is used, and] It is necessary to have the instruction fetch unit of the number corresponding to the number of the instruction styles processed to

juxtaposition, and an instruction decoder, and when the instruction style processed to coincidence is made [many], there is a problem that hardware increases remarkably. In addition, that two or more instruction fetch units can perform an instruction fetch to coincidence shows that the pass to instruction memory is independently prepared to two or more instruction fetch units. Moreover, having the instruction fetch unit of the number corresponding to the number of the instruction styles processed to juxtaposition and an instruction decoder shows that two or more instruction fetch units by Time Division Multiplexing operate, although the pass to instruction memory is one.

[0007] Therefore, the purpose of this invention is little hardware and is offering the information processor which can perform parallel execution of more instruction styles.

[0008]

[Means for Solving the Problem] In order to solve this technical problem, the information processor of this invention is equipped with the instruction fetch section which reads the instruction style of N individual (N> 1) with the gestalt of Time Division Multiplexing, the instruction buffer section, the instruction decoding section, and an execution unit. The instruction fetch section is equipped with the instruction fetch address corresponding to each of the instruction style of N individual of comparatively many numbers, and reads the single or multiple instruction of the instruction style by instruction fetch actuation to one instruction style.

[0009] The instruction buffer section is equipped with M instruction buffers (N>M> 1) of the suitable number to the configuration of two or more below-mentioned instruction execution means, chooses one of the M instruction buffers, stores the instruction which the instruction fetch section read, and outputs one instruction which each of M instruction buffers holds to the instruction decoding

section, respectively. The instruction decoding section is equipped with the M same instruction decoders as the number of an instruction buffer, and resource allocation means, each M instruction decoder decodes M instructions supplied from M instruction buffers, respectively, and a resource allocation means inputs the condition of an execution unit, and the decoded instruction which M instruction decoders output at least, and supplies the decoded instruction in which resource allocation is possible to an execution unit.

[0010] An execution unit has two or more instruction execution means, and calculates according to the decoded instruction which is outputted from the instruction decoding section. According to this configuration, by the instruction buffers and instruction decoders of an instruction style fewer than a number, it becomes possible to process many instruction styles, and parallel execution of more instruction styles can be performed by little hardware.

[0011]

[Embodiment of the Invention] Hereafter, the case where four instruction styles are processed about the gestalt of operation of this invention using two instruction buffers, two instruction decoders, and three instruction execution means using drawing 1 and drawing 2 is explained. Drawing 1 shows the block diagram of the data processor which constitutes the information processor in the gestalt of operation of the 1st of this invention. For 100, as for the instruction buffer section and 300, in drawing 1 , the instruction fetch section and 200 are [the instruction decoding section and 400] execution units.

[0012] The instruction fetch section 100 is equipped with the four instruction fetch addresses 110,120,130,140, chooses one from the four instruction fetch addresses in order, outputs it to the instruction address 10, and fetches an instruction from memory (not shown). 20 is the

instruction by which the fetch was carried out. In addition, the difference between the instruction fetch section 100 and the instruction fetch unit 101,102 of the conventional example is as follows. That is, the instruction fetch section 100 is different in that the instruction fetch unit 101,102 multiplexes only a part for the instruction fetch address part of one instruction fetch unit even for the instruction fetch address respectively required for an instruction fetch to being the prepared perfect instruction fetch unit every.

[0013] The instruction buffer section 200 is equipped with two instruction buffers 210,220, and stores alternatively in one of the instruction buffers 210 or 220 the instruction 20 by which the fetch was carried out from memory to two or more instruction styles. Therefore, an instruction buffer 210,220 holds the instruction of an instruction style different, respectively. Furthermore, the oldest instruction that each instruction buffer 210,220 stores is outputted to the instruction decoding section 300. In addition, the above-mentioned instruction buffer 210,220 has composition like a FIFO memory respectively.

[0014] The instruction decoding section 300 is equipped with two instruction decoders 310,320 and resource allocation means 330, and two instruction decoders 310,320 decode independently the instructions 30 and 40 of an instruction style different, respectively which the instruction buffer 210,220 of the instruction buffer section 200 outputs, respectively. Moreover, the resource allocation means 330 considers the instruction execution condition 70 which shows the running state of the decoded instructions 50 and 60 which an instruction decoder 310,320 outputs, and an execution unit 400 as an input, judges whether instruction 50 or instruction 60 can be executed in an execution unit 400, and publishes the instruction which can be executed by control information (instruction issue)

80 to an execution unit 400. If it explains concretely, instructions 50 and 60 will always be outputted to the execution unit 400, and the resource allocation means 330 will control by control information 80 which instruction is published to which instruction execution means 410-430.

[0015] 90 is control information (instruction fetch), directs control of renewal of the instruction fetch address to the instruction fetch section 100, and directs nullification by the number [finishing / instruction decoding] (consumption cutting tool) or branching to the instruction buffer section 200. An execution unit 400 has two or more instruction execution means. With the gestalt of this operation, it has three instruction execution means 410,420,430. In this, the instruction execution means 410 is the 1st integer-arithmetic unit, the instruction execution means 420 is the 2nd integer-arithmetic unit, and the instruction execution means 430 presupposes that it is for example, a load store unit. The instruction execution means 410,420 which is an integer-arithmetic unit will perform the integer arithmetic according to the decoded instruction, if the instruction decoded, respectively is received. Moreover, the instruction execution means 430 which is a load store unit will perform load actuation or store actuation to memory according to the decoded instruction, if the decoded instruction is received. Here, activation latency of a load store unit is taken as a two cycle.

[0016] With the information processor constituted as mentioned above, four instruction styles are performed to juxtaposition. Hereafter, the actuation is explained using drawing 2 . Drawing 2 is the timing chart of the information processor in the gestalt of this operation of operation. The same number is given to the thing same for explanation as drawing 1 . In drawing 2 , it is shown that a notation called A3 is the information about the 3rd

instruction of the instruction style A. Moreover, it is shown that a notation called A2-4 is the information about the 2nd to the 4th instruction of the instruction style A.

[0017] Therefore, in the cycle 1 of drawing 2 , the instruction fetch addresses 110,120,130 and 140 show that the address of the 0th instruction of the instruction styles A, B, C, and D is held, respectively. Hereafter, actuation is explained for every cycle.

: (Cycle 1) The instruction fetch section 100 chooses the instruction fetch address 110 of the instruction style A from the instruction fetch addresses 110,120,130 and 140, outputs it to the instruction address 10, and performs an instruction fetch. The read instruction is inputted from instruction 20. At this time, the 0th to the 1st two instructions of the instruction style A are read to the instruction 20. Although processing in connection with the instruction by which the fetch was originally carried out before the cycle 1 is performed, since actuation is clarified by next explanation, the instruction buffer section 200, the instruction decoding section 300, and the instruction-execution section 400 are not explained here.

[0018] : (Cycle 2) The instruction fetch section 100 chooses the instruction fetch address 120 of the instruction style B, outputs it to the instruction address 10, and performs an instruction fetch. The 0th to the 1st two instructions of the instruction style B are read to the instruction 20. The instruction buffer section 200 stores in an instruction buffer 210 the 0th to the 1st instruction of the instruction style A fetched in the cycle 1, and outputs the 0th instruction of the instruction style A to the instruction decoder section 300 from an instruction buffer 210. In the instruction decoder section 300, an instruction decoder 310 receives and decodes the 0th instruction of the instruction style A from instruction 30. Moreover, the information that there is no effective

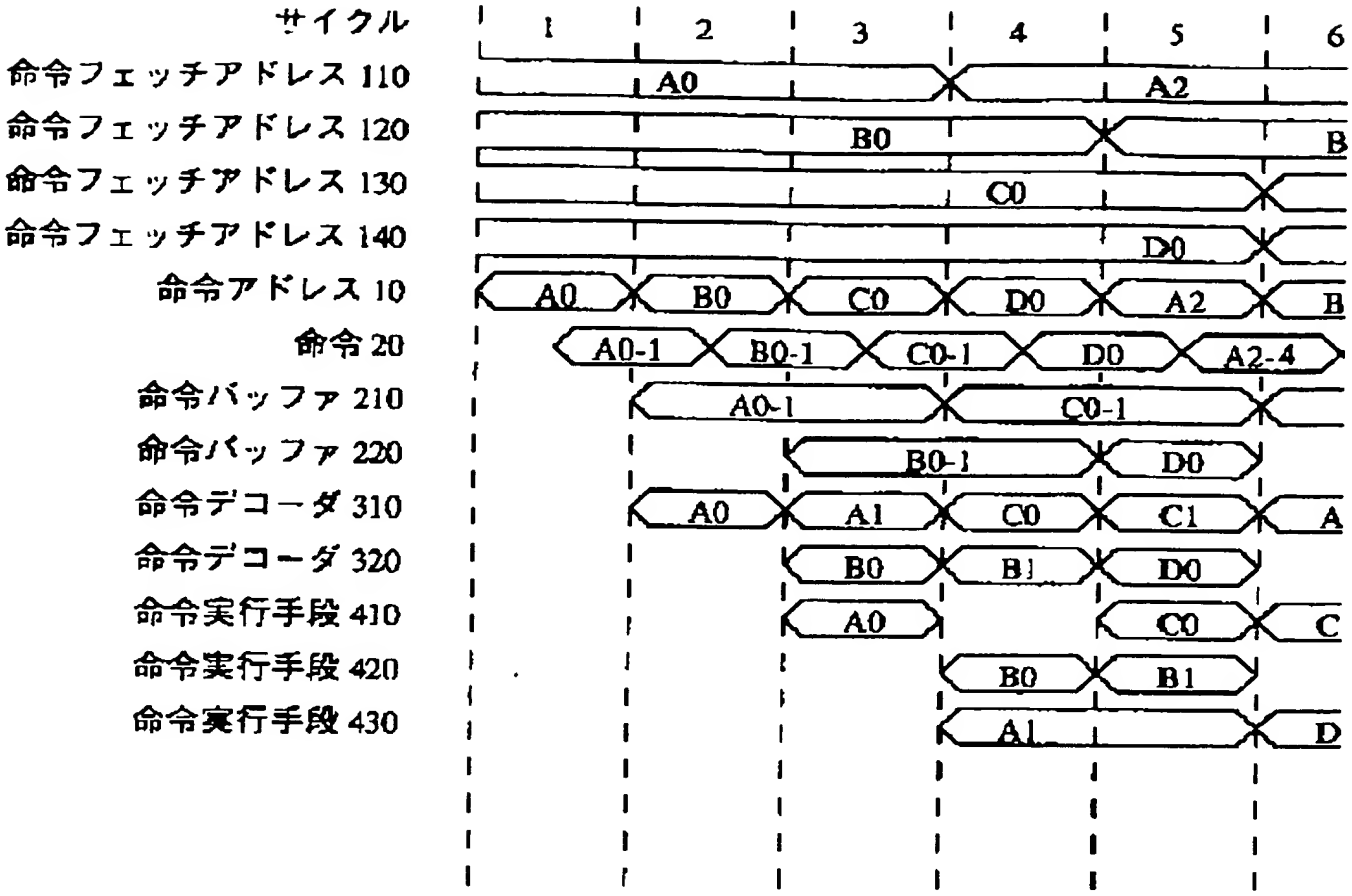
instruction in instruction 40 is outputted, and an instruction decoder 320 decodes the information. Furthermore, the resource allocation means 330 processes the output of instruction decoders 310 and 320. Here, since it being an idle state and the output of an instruction decoder 320 are invalid instructions, an execution unit 400 operates so that the output of an instruction decoder 310 may be assigned to the instruction-execution means 410.

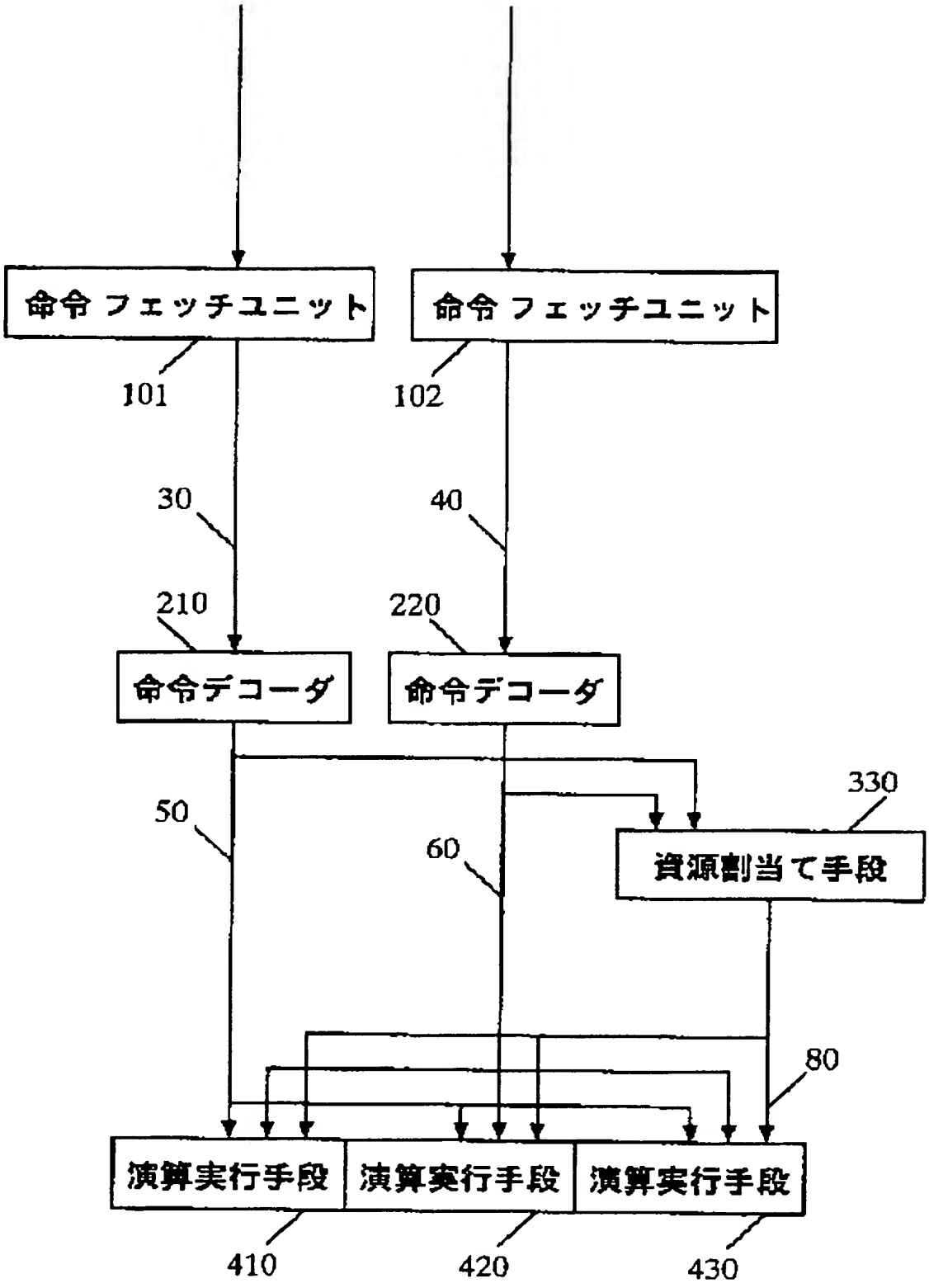
[0019] : (Cycle 3) The instruction fetch section 100 chooses the instruction fetch address 130 of the instruction style C, outputs it to the instruction address 10, and performs an instruction fetch. The 0th to the 1st two instructions of the instruction style C are read to the instruction 20. The instruction buffer section 200 stores in an instruction buffer 220 the 0th to the 1st instruction of the instruction style B fetched in the cycle 2, outputs the 1st instruction of the instruction style A to the instruction decoder section 300 from an instruction buffer 210, and outputs the 0th instruction of the instruction style B to the instruction decoder section 300 from an instruction buffer 220. In the instruction decoder section 300, an instruction decoder 310 receives and decodes the 1st instruction of the instruction style A from instruction 30, and from instruction 40, an instruction decoder 320 receives the 0th instruction of the instruction style B, and it decodes it. The resource allocation means 330 operates so that the output of an instruction decoder 310 may be assigned to the instruction-execution means 410 and it may assign the output of an instruction decoder 320 to the instruction-execution means 420 from the condition of an execution unit 400, and the output of instruction decoders 310 and 320. In the instruction-execution section 400, the instruction-execution means 410 executes the 0th assigned instruction of the instruction style A.

[0020] : (Cycle 4) The instruction fetch section 100

chooses the instruction fetch address 140 of the instruction style D, outputs it to the instruction address 10, and performs an instruction fetch. The 0th instruction of the instruction style D is read to the instruction 20. The instruction buffer section 200 stores in an instruction buffer 210 the 0th to the 1st instruction of the instruction style C fetched in the cycle 3, outputs the 0th instruction of the instruction style C to the instruction decoder section 300 from an instruction buffer 210, and outputs the 1st instruction of the instruction style B to the instruction decoder section 300 from an instruction buffer 220. In the instruction decoder section 300, an instruction decoder 310 receives and decodes the 0th instruction of the instruction style C from instruction 30, and from instruction 40, an instruction decoder 320 receives the 1st instruction of the instruction style B, and it decodes it. The resource allocation means 330 operates so that the output of an instruction decoder 310 may be assigned to the instruction-execution means 430 and it may assign the output of an instruction decoder 320 to the instruction-execution means 420 from the condition of an execution unit 400, and the output of instruction decoders 310 and 320. In the instruction-execution section 400, the instruction-execution means 420 executes the 1st instruction of the instruction style A with which the 0th assigned instruction of the instruction style B was assigned to the instruction-execution means 430.

[0021]


サイクル
1
2
3
4
5
6
命令フェッチアドレス 110
A0
A2
命令フェッチアドレス 120
B0
B
命令フェッチアドレス 130
C0
命令フェッチアドレス 140
D0
命令アドレス 10
A0
B0
C0
D0
A2
B
命令 20
A0-1
B0-1
C0-1
D0
A2-4
命令バッファ 210
A0-1
C0-1
命令バッファ 220
B0-1
D0
命令デコーダ 310
A0
A1
C0
C1
A
命令デコーダ 320
B0
B1
D0
命令実行手段 410
A0
C0
C
命令実行手段 420
B0
B1
命令実行手段 430
A1
D


命令フェッチユニット
命令フェッチユニット
101
102
30
40
210
220
命令デコーダ
命令デコーダ
50
60
330
資源割当て手段
80
演算実行手段
演算実行手段
演算実行手段
410
420
430

This Page is Inserted by IFW Indexing and Scanning Operations and is not part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images include but are not limited to the items checked:

- ☐ BLACK BORDERS
- ☐ IMAGE CUT OFF AT TOP, BOTTOM OR SIDES
- ☑ FADED TEXT OR DRAWING
- ☐ BLURRED OR ILLEGIBLE TEXT OR DRAWING
- ☐ SKEWED/SLANTED IMAGES
- ☐ COLOR OR BLACK AND WHITE PHOTOGRAPHS
- ☐ GRAY SCALE DOCUMENTS
- ☐ LINES OR MARKS ON ORIGINAL DOCUMENT
- ☐ REFERENCE(S) OR EXHIBIT(S) SUBMITTED ARE POOR QUALITY
- ☐ OTHER: _____

IMAGES ARE BEST AVAILABLE COPY.
As rescanning these documents will not correct the image problems checked, please do not report these problems to the IFW Image Problem Mailbox.